A Unifying Data Framework for Facilities Design Algorithms

Marc Goetschalckx

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia, 30332-0205

ABSTRACT

Currently, there exist many different representations and corresponding data structures for departments in facilities design algorithms. These data structures were developed independently and at different times to satisfy the specific needs of the various design algorithms. As a consequence, layouts generated by different facilities design algorithms are very difficult to compare and the algorithms themselves are executed in isolation.

We will present a general purpose data framework that can be used by a large variety of facilities design algorithms. These algorithms can be area based or graph based, have discrete sized or continuous sized departments, have or lack aisle networks, and can have different flow sets. This general data framework will allow the comparison of layouts created by different algorithms and the user directed execution of several algorithms for the same design problem from a common data base. We will illustrate the use of this data framework with several classical and recent algorithms.

1. INTRODUCTION

Facilities design is concerned with designing the appropriate physical environment for an activity to best support this activity. This activity can be in manufacturing, health care, service, education, or administration. The layout of the physical environment, more commonly called facility, can have a significant impact on the operating cost and flexibility of the activity. Because facilities design is a complex and fuzzy problem with ill defined objectives and constraints, designers have long been using computer algorithms to assist them in designing and evaluating layouts.

At the current time, there does not exist an formulation and corresponding solution algorithm that has been able to capture even a large part of the complexity of the facilities design problem. Researchers have made various simplifying assumptions to reduce the complexity of the problem to a manageable level. Using these different assumptions, the design algorithms created in turn layouts which of vastly different types.

A prime example is the graphical and area based design algorithms. Graphical algorithms generate a relationship diagram which shows the relative position of the various departments in the layout. The quality of the relationship diagram is usually measured by its adjacency score. Area based algorithms generate a conceptual block layout. The quality of the block layout is usually measured by its distance score. It is nearly impossible to compare the quality of an adjacency graph with the quality of a conceptual block layout. Moreover, how to convert an adjacency graph into a high quality block layout is still an open research question. How can the designer then select the appropriate algorithm for his application?

Material handling networks, that contain elements such as aisles and doors and pickup and deposit stations, make the problem even more complex. Similarly, the consideration of different material flows in the facility, such as people, products, and containers, complicates the problem significantly.

The research presented in this paper attempts not so much to develop algorithms for creation of better adjacency graphs or conceptual block layouts, but rather attempts to provide an underlying framework so that different algorithms can be used for a common data set. The paradigm is similar to the operation of a spreadsheet, where data are read in, operations performed, and the results are again stored in the same spreadsheet. The benefit to the practitioner will be the ability to use different algorithms in succession in the hope that the final layout will be better suitable for his application. Researchers will be able to develop better algorithms for specific design tasks while being assured that a final layout can be generated and will be able to better evaluate and compare algorithm quality.

2. REVIEW AND CLASSIFICATION OF FACILITIES DESIGN ALGORITHMS

In this section, an attempt is made to catalogue and classify all currently available facilities design algorithms. The objective is to identify for each algorithm what are the required and optional data items and what are the algorithm results and how are the generated designs evaluated. The union of all the required data items will then provide a starting point for the unified facilities design data framework. While a strong effort was made to be complete, some algorithms might not have been included. It is the intention of the author to construct a site on the world-wide-web Internet containing the following catalogue and a data entry form so that new algorithms and their data requirements, generated solutions, and evaluation methods can reported.

A classification of facilities design models and algorithms is shown in the next Figure. One major distinguishing feature is the presence of the areas of the departments. Graphical algorithms do not include the area of the departments and as such can only provide information on the relative position of the department in the overall layout. Area algorithms on the other hand do include the areas of the departments and thus can create a true conceptual block layout.

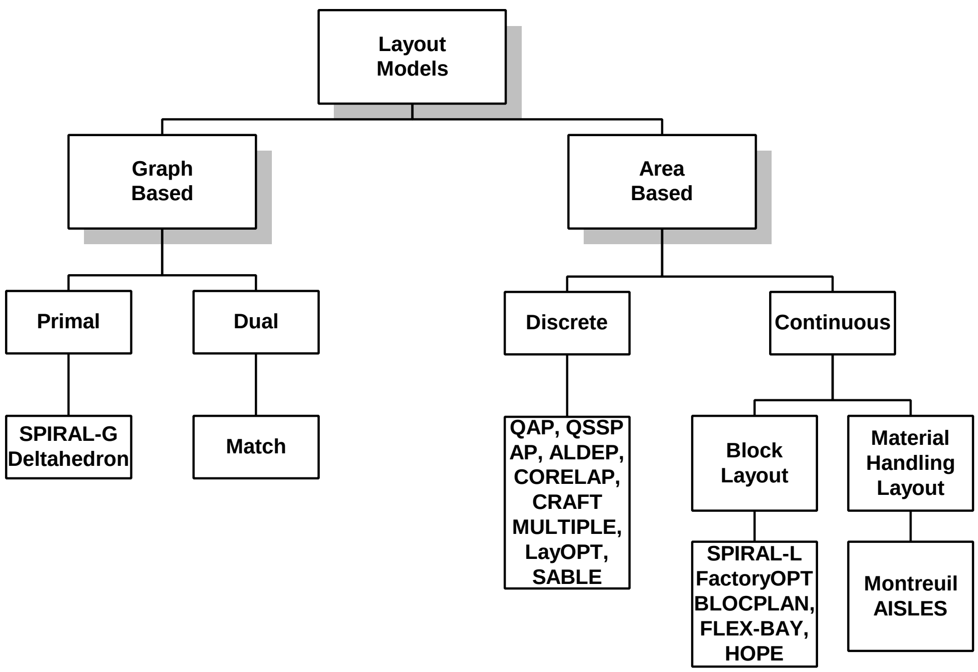

Figure 1. Layout Models and Algorithms Classification

A graph representing the relative position of the departments can only be converted to a conceptual block layout if it is planar, i.e., when it can be drawn in a two dimensional plane without crossing edges between adjacent departments. The graph based algorithms can be further divided into primal, feasible algorithms which at all times maintain a planar graph, and dual algorithms which optimize the adjacency objective function but do not necessarily yield a planar, feasible graph.

The area-based algorithms can then be further divided into discrete and continuous algorithms. In discrete area models the departments and building are composed of a integer number of equal sized unit squares. In continuous area models the dimensions of the building and departments can have fractional values. Probably the most famous of all discrete area algorithms is CRAFT, Armour and Buffa (1964). A recent discrete area algorithm based on space filling curves is MULTIPLE, Bozer et al (1994). A heuristic algorithm to create a continuous layout is given by the layout phase of SPIRAL, Goetschalckx (1991), while a formulation to find the optimal continuous layout for perimeter constrained departments is given in Montreuil (1990).

Finally, area based algorithms can explicitly incorporate the material handling aisles network and the interface points between departments and the aisles network. An optimal formulation for this problem was given by Montreuil (1990) and an heuristic method, called AISLES and based on hexagonal adjacency graphs, was given in Goetschalckx and Palliyil (1994).

There exists an extensive body of research on various algorithms to create either adjacency graphs or conceptual layouts. It was not our goal to collect and review these algorithms, but rather to provide a classification of their data requirements and final results. A comprehensive collection of references for facilities design and layout is given in Meller (1995).

3. UNIFIED DATA FRAMEWORK

The objective of the unified data framework for facilities design is to provide a data base structure for facilities design projects so that all algorithms can extract all their required data from the data base instance for a project and can deposit their results completely back into the database. A particular algorithm does not have to use all elements of the data available, nor does it have to determine values for all the decision variables. For instance, graph based algorithms most likely will not use the department areas as data and will not determine the location of the department area.

Based on the above list of algorithms, the unified data framework has to contain the following fundamental elements:

1. activity centers (more commonly called departments),
2. activity relationships for more than one type of material flow,
3. boundary conditions (such as building and site constraints),
4. material handling networks (such as aisles, doors, and elevators), and
5. evaluation methods.

Each of these elements will be discussed in further detail.

3.1. Activity Centers

The common name for an activity center is a department. Determining the correct number and grouping of activity centers in a layout project is essential for the final success of the generated layout. An activity center is defined to be atomic, has homogeneous material flow relationships with other activity centers, and has clearly defined interface points with the material handling network. It can have additional characteristics such as a maximum allowable shape factor and the restriction that it is fixed in place.

3.1.1. General Department Data

The basic data elements associated with a department are given next. Even though the layout enclosure or building is not an activity center, its data structure is identical to the data structure of activity centers. Since the building provides data to validate the data of the departments, its data record is always the first data record in the list of activity data records.

3.1.1.1. Label

Each department is identified by a label. The label of a department can only consist of alphanumeric characters and underscores. Blanks are not allowed. The minimum label length is one character, the maximum label length is 15 characters.

3.1.1.2. Name

3.1.1.3. Required Area

The required department area is a positive quantity. The required area of a department can be a fractional, i.e., real number. The required area is denoted by A_i.

3.1.2. Atomic Area

An activity center is assumed to be atomic, i.e., it cannot be divided or overlapped by other activity centers and material handling networks can not traverse an activity center. An activity center occupies a contiguous area in the layout. The boundary of the activity center is formed by a sequence of connected perpendicular horizontal and vertical lines. This definition allows "U" and "L" shapes as well as the more traditional rectangular activity center shapes. It also accommodates the activity center shapes created by improvement schemes for departments composed of unit squares such as the CRAFT algorithm. It should be observed that the resulting shapes are not necessarily convex nor compact. Some examples of possible activity center shapes are given in the next figure.

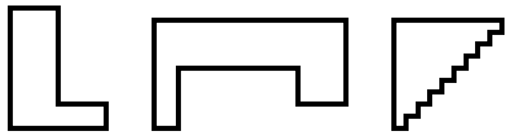

Figure 2. Possible Activity Center Shapes

For each activity center the following data are stored in the data framework:

3.1.2.1. Number of Vertices

The number of vertices can be zero if the department location has not yet been determined. If the department has a location then the minimum number of vertices is four since the boundary of the department consists of perpendicular line segments. A square or rectangular department has four vertices. The building always has a four or more vertices.

3.1.2.2. Realized Area

After a layout algorithm has determined the location of each department, the actual department area can be computed. The actual area is denoted by Ω_i.

3.1.2.3. Realized Perimeter

After a layout algorithm has determined the location of each department, the actual department perimeter can be computed. The actual perimeter is denoted by Π_i.

For each department a list of vertices is stored which contains the coordinates of the vertices of the department.

3.1.2.4. Coordinates

The coordinates of any point, be it either a vertex or an interface point, are given by an a x and y coordinate. The lower left (south west) corner of the building has by definition the coordinates (0, 0). X coordinates are measured from this origin and are increasing to the right (east). Y coordinates are measured from this origin and are increasing to the top (north). Coordinates can be fractional, i.e., real numbers.

3.1.3. Homogeneous Material Flows

An activity center is such that it has a homogeneous material flow with all other activity centers in the layout. For example, the slow moving “C” items and the fast moving “A” items in a warehouse have different material flow characteristics and their storage areas should be considered as different activity centers.

3.1.4. Material Flow Interface Points

An interface point is the location where an activity center exchanges material with either another activity center or the material handling network. While the first two properties of activity centers are especially important during the construction of the adjacency graph and the conceptual block layout, interface points with the material handling network are most important during the determination of the material handling layout. An activity center has one or more interface points and these interface points are always located inside the department area or at the boundary of the activity center. Typical examples of material flow interface points are doors, truck docks, and pickup and deposit stations. Interface points can be either for outgoing, incoming, or incoming and outgoing material flow. The type of the interface point is then pick up, deposit, or combined, respectively.

For each activity center, the number of material flow interface points is included in the data framework.

3.1.4.1. Number of Interface Points

This number can be zero if no interface points have been determined or are required. Otherwise, this number must be a positive integer number.

For each activity center and for each of its interface points, the following data elements are present in the data base framework:

3.1.4.2. Coordinates

Identical requirements to the coordinates of a vertex of a department.

3.1.4.3. Type

One of three possible types: PICKUP, DEPOSIT, or COMBINED.

3.1.5. Additional Characteristics

A activity center can have additional characteristics. The following characteristics are included in the data framework:

3.1.5.1. Minimum Allowable Width

The minimum allowable width specifies the lower bound on the x dimension of a department. This lower bound can be zero.

3.1.5.2. Maximum Allowable Width

The maximum allowable width specifies the upper bound on the x dimension of a department. This upper bound can be at most the building width.

3.1.5.3. Minimum Allowable Depth

The minimum allowable depth specifies the lower bound on the y dimension of a department. This lower bound can be zero.

3.1.5.4. Maximum Allowable Depth

The maximum allowable depth specifies the upper bound on the y dimension of a department. This upper bound can be at most the building depth.

3.1.5.5. Maximum Allowable Shape Ratio

Traditionally, the maximum allowable shape ratio has been the maximum of the ratios of the length over width and width over length of the convex hull enclosing the department. For a rectangular department this reduces to the maximum of the ratios of length over width or width over length, or

$$R_i = \max\left\{\frac{L_i}{W_i}, \frac{W_i}{L_i}\right\}$$

where L and W are the length and width of the enclosing convex hull, respectively. For example the shape ratio of a square department is 1.00 and the shape ratio of a four by one department is 4.00.

3.1.5.6. Maximum Allowable Shape Complexity

Bozer et al. (1994) proposed a shape complexity measure based on the ratio of the department perimeter over the smallest perimeter of a department with orthogonal boundary segments that can enclose the same area, or

$$S_i = \frac{\Pi_i}{4\sqrt{A_i}}$$

where Π is the realized department perimeter and A is the required department area, respectively. For example the shape complexity factor of a square department is 1.00 and the shape complexity factor of a four by one department is 1.25.

3.1.5.7. Area Penalty

A penalty factor in the objective function for every unit of area the department is different than its required area. The penalty for a department to be too small is positive. In most cases, the penalty for a department to be too large is equal to zero, but the penalty for an oversized department can be negative. Let m_i be the unit area penalty for department *i*, then the objective function penalty is:

$$m_i \cdot \left[A_i - \Omega_i \right]$$

3.1.5.8. Shape Penalty

A penalty factor in the objective function. If the department has a shape which exceeds the maximum allowable shape ratio of the maximum allowable shape complexity, then the objective function is penalized by the following term, where n_i denotes the shape penalty and Φ_i denotes the realized shape ratio:

$$n_i \cdot \left[R_i - \Phi_i \right]$$

3.1.5.9. Fixed in Place

The department can be either: MOVEABLE or FIXED.

If the current position of the department in the layout cannot be changed by the algorithm then the department is said to be fixed in place. Examples are departments with heavy machinery or specialized foundations for which the cost of moving would be prohibitive.

3.1.6. Display Characteristics

Finally, each activity center has a number of display characteristics that are only used to represent the department on the computer screen or in reports.

3.1.6.1. Area Color

This color is used to draw the department area as determined by the sequence of its vertices.

3.1.6.2. Boundary Color

This color is used to draw the outside boundary of the department as determined by the sequence of its vertices.

3.1.6.3. Node Color

This color is used to draw the size independent symbol for the department when the algorithm result is an adjacency graph.

Since these colors and their representation depend strongly on the computer system and program executing the algorithm, several formats are possible. The first format restricts the colors to a list of color names. The second format gives the colors with an integer color index.

Table 1. Valid Color Names

BLACK	DARKGRAY
BLUE	NAVY
GREEN	FOREST
CYAN	OCEAN
RED	BROWN
MAGENTA	PURPLE
YELLOW	OLIVE
WHITE	GRAY

3.2. Multiple Types of Activity Relations

The activity relation expresses the desirability of two activity centers or departments to be close to each other. Positive relations indicate that departments have a mutual attraction to each other, negative relations denote rejection between a pair of departments. Often positive activity relations are based on some measure of material flow between the two departments. Negative activity relations are based on environmental incompatibilities such as noise, vibration, or pollution. The collection of all binary activity relationships is called a relationship matrix. A layout can have multiple relationship matrices for different types of "material". For example, in hospital design, there are material flows corresponding to personnel, patients, visitors, drugs, and supplies. The collection of all relationship matrices in a project is called a relationship tensor. The relationship between departments and *i* and *j* for material *p* is denoted by f_{ijp}. The cost moving this material one distance unit is given by c_{ijp}.

3.2.1. General Commodity Data

The following scalar data item are associated with the activity relationships:

3.2.1.1. Number of Commodities

For each commodity the following data items are stored in the data framework:

3.2.1.2. Label

The commodity label has the same restrictions and conventions as the department label.

3.2.1.3. Name

The commodity name has the same conventions and restrictions as the department name.

3.2.2. Relationship Data

The relationships between the department pairs are then given by the following data records.

3.2.2.1. From Department

The label of the origin department.

3.2.2.2. To Department

The label of the destination department.

3.2.2.3. Commodity

The commodity label

3.2.2.4. Relationship

The asymmetrical relationship f_{ijp}. This relationship can be positive to indicate that it is desirable that these departments are located close to each other, or negative to indicate that there exists an incompatibility between the departments.

The relationships are asymmetrical, i.e., f_{ijp} and c_{ijp} denote a different relationship and associated cost than f_{jip} and c_{jip}. Algorithms that are based on symmetrical relationship data should first convert the asymmetrical data to a symmetrical format. This is usually done by adding the corresponding asymmetrical relationships together to form one symmetrical relationship.

3.2.2.5. Unit Transportation Cost

The asymmetrical unit transportation cost c_{ijp}.

3.3. Material Handling Networks

The material handling network(s) is represented in the same manner as the department by a sequence of vertices. Most of the time there exists only a single connected material handling network, but the data framework allows for the possibility of more than one disjointed networks.

3.3.1. General Material Handling Networks Data

The following data are stored in the data framework:

3.3.1.1. Number of Networks

This number can be zero is not networks have designed, otherwise it is a positive integer number.

3.3.2. Atomic Area

For every network the following data is stored in the data framework:

3.3.2.1. Number of Vertices

A positive number of vertices larger than or equal to four.

3.3.2.2. Boundary Color

Identical requirements to the boundary color of departments

3.3.2.3. Area Color

Identical requirements to the area color of departments

3.3.2.4. Coordinates

Identical requirements to the coordinates of a vertex of a department.

3.3.2.5. Unit Area Cost

In order to compute a total material handling system cost, the cost per unit area of the material handling network also has to be specified. This cost is the annualized cost for building and maintaining a square unit of the material handling network and then reduced to the same planning horizon as the material flow relationships. The unit area cost allows the differentiation between narrow and wide aisles.

3.3.3. Material Flow Interface Points

An interface point is the location where a material handling network exchanges material with either an activity center or another material handling network. A material handling network has one or more interface points and these interface points are always located inside the network area or at the boundary of the network. Typical examples of material flow interface points are doors, truck docks, and pickup and deposit stations. Interface points can be either for outgoing, incoming, or incoming and outgoing material flow. The type of the interface point is then pick up, deposit, or combined, respectively.

For each material handling network, the number of material flow interface points is included in the data framework.

3.3.3.1. Number of Interface Points

This number can be zero if no interface points have been determined or are required. Otherwise, this number must be a positive integer number.

For each material handling network and for each of its interface points, the following data elements are present in the data base framework:

3.3.3.2. Coordinates

Identical requirements to the coordinates of a vertex of a department.

3.3.3.3. Type

One of three possible types: PICKUP, DEPOSIT, or COMBINED.

3.4. Evaluation Methods

The data framework stores the results of three ways to evaluate the computed layout and one way to evaluate the adjacency graph.

3.4.1. Adjacency

<u>3.4.1.1. Adjacency Score</u>

Departments are considered adjacent if they have common boundary of non-zero length. A adjacency matrix can be constructed with elements a_{ij}, that are equal to one for every pair of adjacent departments and equal to zero otherwise. The total adjacency score is then the sum of products of the corresponding elements in the adjacency matrix and the relationship matrix, or:

$$Adjacency = \sum_p \sum_i \sum_j a_{ij} f_{ijp}$$

The adjacency graph is evaluated in a similar fashion. The adjacency matrix represents in this case the adjacency of the departments in the adjacency graph.

3.4.2. Distance

<u>3.4.2.1. Centroid-To-Centroid Distance Cost</u>

If no material handling network have been designed, the quality of the layout can be estimated by the rectilinear centroid to centroid distance cost. The centroid (xc_i, yc_i) of each department can be computed. A centroid distance matrix can be constructed with elements dc_{ij}, that are equal to the rectilinear centroid to centroid distance between every pair of departments. The total distance cost is then the sum of products of the corresponding elements in the adjacency matrix and the relationship matrix and the unit transportation cost matrix, or:

$$Centroid\ Distance = \sum_p \sum_i \sum_j dc_{ij} c_{ijp} f_{ijp}$$

Observe that this performance measure does not include relationships with the outside perimeter or building.

<u>3.4.2.2. Material Handling Distance Cost</u>

If material handling network have been designed, then the total amount of material handling can be more accurately estimated by the material handling distance cost. A material handling distance matrix can be constructed with elements dn_{ij}, that are equal to the distance over the material handling network between every pair of departments. The total material handling distance cost is then the sum of products of the corresponding elements in the adjacency matrix and the relationship matrix and the unit transportation cost matrix, or:

$$\text{Network Distance} = \sum_p \sum_i \sum_j dn_{ij} c_{ijp} f_{ijp}$$

Observe that this performance measure does include relationships with the outside perimeter or building since the interface points with the outside have been determined.

3.4.3. Total Cost

In order to compute a total material handling systems cost, several additional cost parameters have to be specified.

3.4.3.1. Pickup Interface Point Cost

The annualized cost of establishing a pickup interface point and then reduced to the time horizon compatible with the flow relationships, e.g., divided by 52 for weekly flows. By convention a pickup station is used to pick up material from a department.

3.4.3.2. Deposit Interface Point Cost

The annualized cost of establishing a deposit interface point and then reduced to the time horizon compatible with the flow relationships, e.g., divided by 52 for weekly flows. By convention a deposit station is used to pick up material from a department.

3.4.3.3. Combined Interface Point Cost

The annualized cost of establishing a combined interface point and then reduced to the time horizon compatible with the flow relationships, e.g., divided by 52 for weekly flows.

3.4.3.4. Total Cost

The total cost of a facilities design is the sum of the material handling distance cost, the cost of establishing all the interface points and the cost of building the material handling network. This total cost can only be computed if the material handling network has been determined, otherwise it is equal to zero.

4. DATA FRAMEWORK ILLUSTRATIONS

We will show for a number of contemporary algorithms that the data framework contains all the necessary data for the execution of the algorithm and for the storage of the algorithm result. The algorithms have been selected because they have different types of data requirements and different types of results and as such to illustrate the general purpose of the data framework.

4.1. MULTIPLE Algorithm

The MULTIPLE algorithm is a discrete area based improvement algorithm. The unit squares of a department are located following a space filling curve in the layout. The algorithm tests improvements by exchanging the location of two departments on the space filling curve. The MULTIPLE algorithm is capable of designing layouts with multiple floors, a feature which is not implemented in the data framework. More information can be bound in Bozer et al. (1994). The commercial LayOPT program, Systems Modeling Corporation (1995), is based in part on the MULTIPLE algorithm. The following Figure illustrates a computer screen of the LayOPT program.

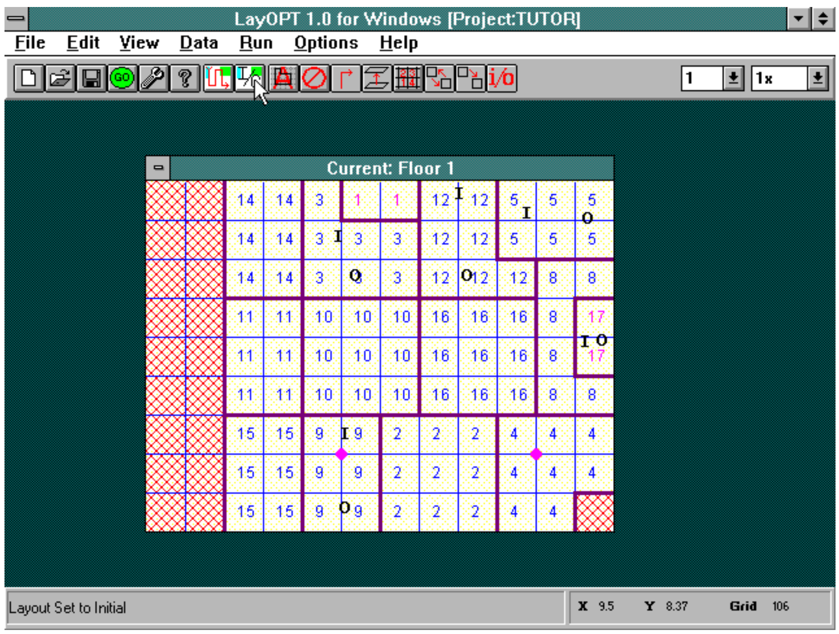

Figure 3. LayOPT Program Illustration

The building dimensions and the department locations and shapes of the above layout can be represented by the vertex data structure of the data framework. Departments that are fixed in place and department input and output points can also be stored. The slack area in the above layout can be represented by a dummy department.

4.2. AISLES Algorithm

The AISLES algorithm constructs a minimum cost aisle based material handling network given a continuous area based conceptual block layout. The AISLES algorithm makes the tradeoff between interface point cost, aisles construction costs, and vehicle transportation costs for various vehicle dispatching policies. It formulates the problem as a large mixed integer programming problem and generates a problem definition file in MPS format, which can be solved by commercial MIP solvers, or determines a network with specialized heuristics. Further information can be found in Goetschalckx and

Palliyil (1994). The following Figure illustrates a computer screen of the AISLES program.

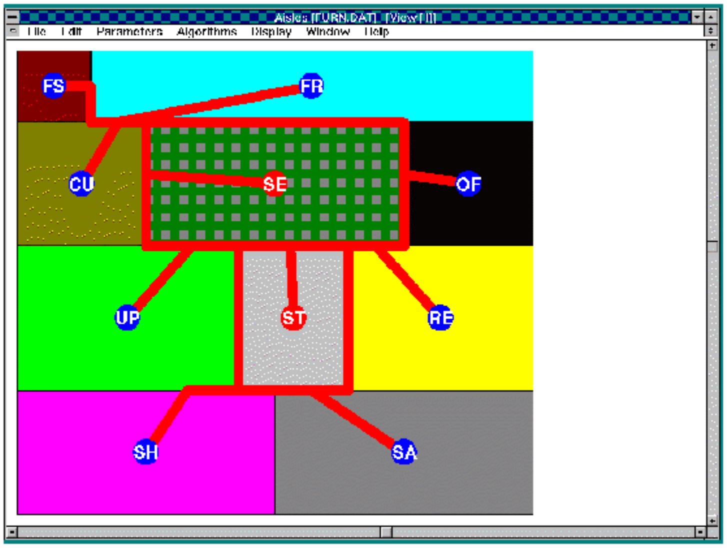

Figure 4. AISLES Screen Illustration

The building dimensions and the given block layout can be stored in the data framework. The designed aisle network can also be stored together with the interface points between the aisle network and the departments. Note that the angled lines from the department centroids to the interface points are not actual aisles but just a graphical representation of the flow from the department to the aisle network. In this case the material handling network distance can be computed.

5. CONCLUSIONS

Up to this time the facilities design methodology was severely handicapped by the lack of a common standard to describe problem data and algorithm results. Such standards exist for other design disciplines such as the MPS standard for linear and mixed integer programming. This made it hard to compare the results and efficiencies of various algorithms and to transfer example problems from one algorithm to another.

This manuscript proposes a unified data framework for the storage of the facilities design problems and resulting layouts. It eliminates many of the above shortcomings and enhance and accelerate the development of new methodology. The framework will adapted to new algorithms and problems as required. An Internet World Wide Web site will be established to hold the latest data framework definition and a collection of test problems.

6. REFERENCES

1. Buffa, E. S., G. C. Armour, and T. E. Vollman, (1964). "Allocating Facilities with CRAFT". ***Harvard Business Review***, Vol. 42, No. 2, pp. 136-158.
2. Bozer, Y. A., R. D. Meller, and S. J. Erlebacher, (1994). "Am Improvement-Type Layout Algorithm for Single and Multiple Floor Facilities." ***Management Science***, Vol. 40. No. 7, pp. 918-932.
3. Goetschalckx, M., (1992). "SPIRAL: An Efficient and Interactive Adjacency Graph Heuristic for Rapid Prototyping of Facilities Design", **European Journal of Operational Research**, Special Issue on Facilities Design, pp. 304-321.
4. Goetschalckx, M. and G. Palliyil, (1994). "A Comprehensive Model for the Concurrent Determination of Aisles and Load Stations for Aisle-Based Material Handling Systems." In **Developments in Material Handling Research**, pp. 161-188, R. Graves et al. (Eds.), Material Handling Institute of America, Charlotte, NC.
5. Meller, R. D., (1995), "The Facility Layout Problem: A Review of Recent and Emerging Research", Department of Industrial Engineering, Auburn University, Auburn, AL.
6. Montreuil, B., (1991), "A Modeling Framework for Integrating Layout Design and Flow Network Design." In **Progress in Material Handling and Logistics**, Vol. 2, J. White and I. Pence, Eds., Springer-Verlag, pp. 95-116.
7. Systems Modeling Corporation, (1995). "LayOPT Demonstration Program".